

Player Animations 2D Parte 3

### Creando el script HealthBar

Modificar el Script HealthBar.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI; //Componentes Interfaz Gráfica

public class HealthBar : MonoBehaviour
{
    [SerializeField]
    public Player character; //Referencia al jugador
    public Image meterImage; //Medidor Meter de la salud
    public Text hpText; //Texto en barra de salud

    void Start()
    {
        character.hitPoints.value = 0;
    }

    void Update()
    {
        if (character != null)
        {
            //Modifica barra de salud
            meterImage.fillAmount = character.hitPoints.value / character.maxHitPoints;
            //Texto a mostrar
            hpText.text = "HP:" + (meterImage.fillAmount * 100);
        }
    }
}
```

→ Es necesario importar el espacio de nombres **UnityEngine.UI** para trabajar con los elementos de la interfaz de usuario.

→ Usamos **[SerializeField]** para ocultar esta propiedad pública en el Inspector. La sintaxis de corchetes para **[SerializeField]** indica que es un atributo. Los atributos permiten comportamientos adicionales a métodos y variables.

→ **meterImage** la creamos esta propiedad por conveniencia y simplicidad, para qué no tener que buscar entre varios objetos secundarios para encontrar la imagen del medidor. Configuraremos esto en el Editor de Unity arrastrando y soltando el objeto **Meter** en esta propiedad, una vez que se ajuste el script **HealthBar**.

→ **hpText** es otra propiedad creada por conveniencia y simplicidad. Bien para establecer en el Editor de Unity arrastrando y soltando el objeto **HPText** en este campo.

→ **Revisa el mismo máximo de puntos de vida no cambió en nuestro diseño actual del juego.**

Player Animations 2D Parte 5

### Modificamos método Start de Player

```
private void Start()
{
    healthBar = Instantiate(healthBarPrefab); //Instanciar HealthBar
    healthBar.character = this; //Referencia del Player en HealthBar
}
```

### Configura el Health Bar

Vuelve al Editor de Unity y selecciona **HealthBar** de la carpeta Prefabs en la vista Proyecto. Modifica las propiedades del Objeto **HealthBar**

Health Bar (Script)

- Meter Image: Meter (Image)
- hp Text: Text (Legacy) (Text)

Player Animations 2D Parte 5

### HealthBar.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI; //Componentes Interfaz Gráfica

public class HealthBar : MonoBehaviour
{
    [SerializeField]
    public Player character; //Referencia al jugador
    public Image meterImage; //Medidor Meter de la salud
    public Text hpText; //Texto en barra de salud

    void Start()
    {
        character.hitPoints.value = 0;
    }

    void Update()
    {
        if (character != null)
        {
            //Modifica barra de salud
            meterImage.fillAmount = character.hitPoints.value / character.maxHitPoints;
            //Texto a mostrar
            hpText.text = "HP:" + (meterImage.fillAmount * 100);
        }
    }
}
```

Player Animations 2D Parte 5

### HealthBar.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI; //Componentes Interfaz Gráfica

public class HealthBar : MonoBehaviour
{
    [SerializeField]
    public Player character; //Referencia al jugador
    public Image meterImage; //Medidor Meter de la salud
    public Text hpText; //Texto en barra de salud

    void Start()
    {
        character.hitPoints.value = 0;
    }

    void Update()
    {
        if (character != null)
        {
            //Modifica barra de salud
            meterImage.fillAmount = character.hitPoints.value / character.maxHitPoints;
            //Texto a mostrar
            hpText.text = "HP:" + (meterImage.fillAmount * 100);
        }
    }
}
```

Player Animations 2D Parte 5

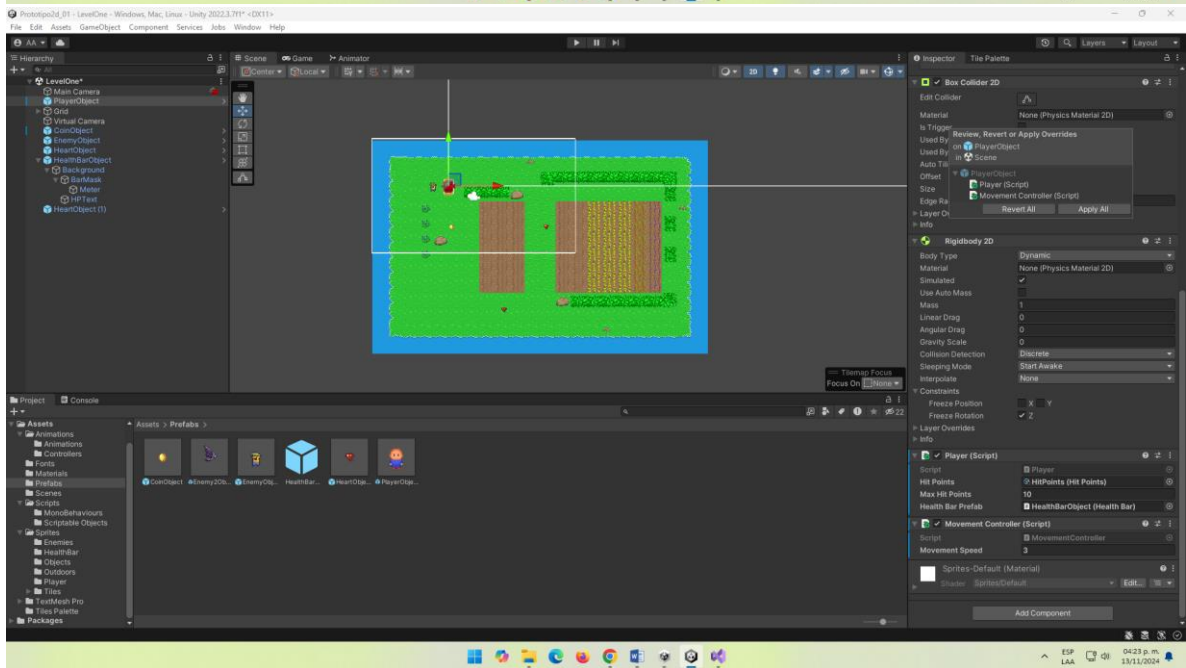
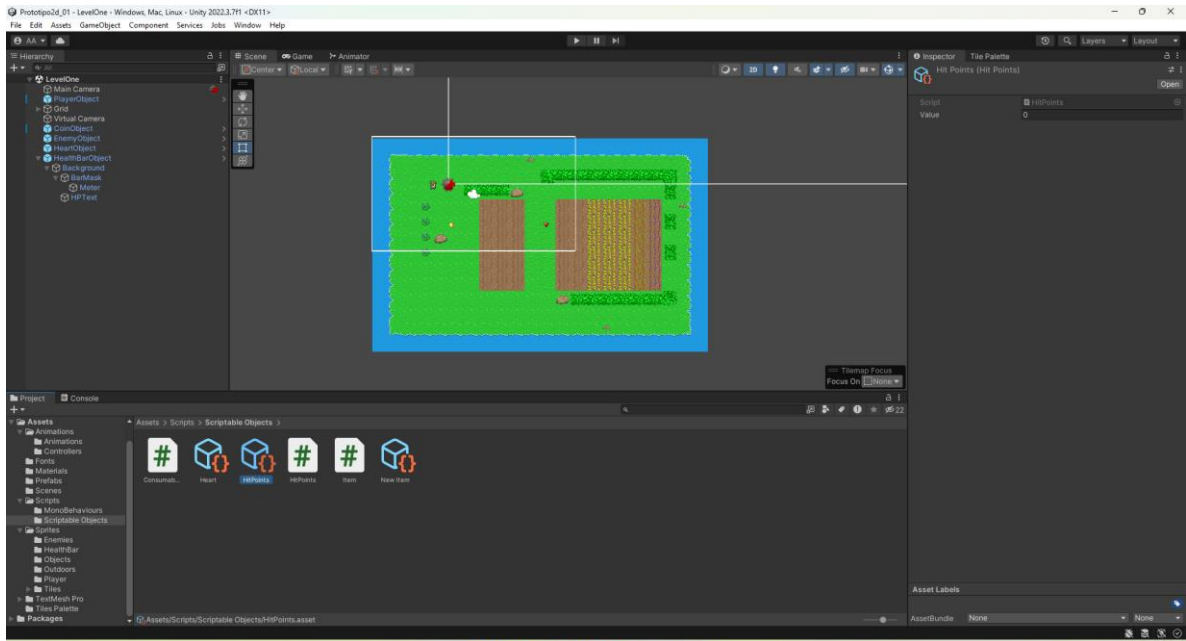
### HealthBar.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI; //Componentes Interfaz Gráfica

public class HealthBar : MonoBehaviour
{
    [SerializeField]
    public Player character; //Referencia al jugador
    public Image meterImage; //Medidor Meter de la salud
    public Text hpText; //Texto en barra de salud

    void Start()
    {
        character.hitPoints.value = 0;
    }

    void Update()
    {
        if (character != null)
        {
            //Modifica barra de salud
            meterImage.fillAmount = character.hitPoints.value / character.maxHitPoints;
            //Texto a mostrar
            hpText.text = "HP:" + (meterImage.fillAmount * 100);
        }
    }
}
```



## Parte 6

